UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P O Box 1450
Alexandria, Virginia 22313-1450
www uspto gov

| APPLICATION NO. | FILING DATE | FIRST NAMED INVENTOR | ATTORNEY DOCKET NO. | CONFIRMATION NO. |
|---|---|---|---|---|
| 10/571,606 | 02/27/2007 | Ingo Meirick | 4147-144 | 9780 |

23117 7590 08/19/2009
NIXON & VANDERHYE, PC
901 NORTH GLEBE ROAD, 11TH FLOOR
ARLINGTON, VA 22203

| EXAMINER | |
|---|---|
| PATEL, MAHENDRA R | |

| ART UNIT | PAPER NUMBER |
|---|---|
| 2617 | |

| MAIL DATE | DELIVERY MODE |
|---|---|
| 08/19/2009 | PAPER |

**Please find below and/or attached an Office communication concerning this application or proceeding.**

The time period for reply, if any, is set in the attached communication.

| Office Action Summary | Application No. 10/571,606 | Applicant(s) MEIRICK ET AL. |
|---|---|---|
| | Examiner MAHENDRA PATEL | Art Unit 2617 |

*-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --*

**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

**Status**

1)☒ Responsive to communication(s) filed on *05/18/2009*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

**Disposition of Claims**

4)☒ Claim(s) *1-29* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-29* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

**Application Papers**

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *10 March 2006* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

**Priority under 35 U.S.C. § 119**

12)☒ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☒ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☒ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

**Attachment(s)**

1) ☒ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☒ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date *02/05/2009, 08/12/2009*.

4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .

5) ☐ Notice of Informal Patent Application

6) ☐ Other: _____.

# DETAILED ACTION

### *Status of the Claims*

This communication is in response to the Amendment filed on 05/18/2009.
Application No: 10/571,606.

Claims 1-29 are pending.
Claims 1-29 are amended by the applicant.

## Response to Amendment

1. An examiner's Response to the record appears below.

2. Applicant's arguments with respect to claims 1-29 have been considered but are moot in view of the new ground(s) of rejection.

### General Note

The Reference characters enclosed within parentheses (e.g. (P (k)), S (K+1)) in Claims 1-29 are treated as an example characters and no special consideration is given during the examination of the claims.

### *Claim Rejections - 35 USC § 103*

1. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

The factual inquiries set forth in *Graham* **v.** *John Deere Co.*, 383 U.S. 1, 148 USPQ 459 (1966), that are applied for establishing a background for determining obviousness under 35 U.S.C. 103(a) are summarized as follows:

1. Determining the scope and contents of the prior art.
2. Ascertaining the differences between the prior art and the claims at issue.
3. Resolving the level of ordinary skill in the pertinent art.
4. Considering objective evidence present in the application indicating obviousness or nonobviousness.

2. **Claims 1-29 are** rejected under 35 U.S.C. 103(a) as being unpatentable over Besset et al. (US 6711126 B1), in view of Yuan et al. (US 6567378 B1).

**Regarding claim 1**, Besset teaches a method of managing a data buffer comprising a queue of consecutive data packets in a base station system of a mobile communications system (**[Col 7, lines 12-14**) (e.g. FIG. 4 shows the **queuing** of CPS packets at the buffer entrance for the case of example, waiting to be multiplexed to form CPS PDUs);

comprising the steps of: said base station system comparing a size of a data packet segment with a size of a next consecutive data packet segment in said buffer (**[Col 2, lines 4-12**] (e.g. an ATM adaptation layer comprises a number of sub-layers: a **service specific segmentation** (i.e. comparing a size of a data packet segment with a size of a next consecutive data packet segment in a buffer ) and reassembly sub layer (SSSAR) forming part of SEG-SSCS which allows data packets exceeding 45 octets **(i.e. size)** to be segmented));

Said base station system identifying said complete data packet based on said comparison **([Col 10, lines 36-42)** (e.g. the **algorithm** (i.e. based on ID comparison) starts by

**reading** the UUI field of a CPS packet being processed (i.e. identifying a data packet). As explained above, this UUI field has the value 27 if at least one other CPS packet follows to complete the AAL2 SDU frame, and the value 26 if, on the contrary, the CPS packet is the last one of the frame. Note that a packet can be both the first and last packet of a frame, as in the case of packet 5 in FIG. 2 (i.e. next data packet segment as a first data packet segment)));

And said base station system discarding said identified complete data from the said buffer ([**Col 11, lines 23-35**) (e.g. If the buffer is saturated, the algorithm follows branch which leads to the step of **discarding the current CPS packet** and setting the parameter PD.sub.AAL2 =TRUE. Consequently, the algorithm for all the following CPS packets up to but excluding the next first packet will then evolve described above to step and thereafter follow branch, whereupon these packets shall be systematically discarded. Thus in the event of a buffer overflow as determined by the decision algorithm for overflow management the current CPS packet and all the following CPS packets of an AAL2 SDU are **systematically discarded even if one or several initial packet(s) had been recorded in the buffer memory**));

Besset differ from the claimed invention in not specifically teaching segments of data packets in a protocol.

However, the preceding limitation is known in the art of communications. In the same field of endeavor, Yuan teaches a method for segmenting data packets ([**Col 1,** lines 10-17] (e.g. Cell relay systems, such as asynchronous transfer mode (ATM) systems, transmit data over a network as a plurality of fixed-length cells. The individual transmissions typically include one or more cells that constitute a portion of variable-length packets used by end systems or

applications. Before transmission, a source station **segments a packet into one or more cells and then transmits the cells** (i.e. A complete data packet is build and transmitted).

Yuan also teaches segments discarding process ([**Col 3,** lines 55-65] (e.g. FIGS. 5A and 5B are flowcharts of **cell discard processing** consistent with the present invention. The processing begins when an output port receives a cell of an incoming packet over a virtual circuit at queuing buffer. Each packet has an **identifier** such as a Virtual Circuit Identifier (VCI), for example, to identify the virtual circuit over which it is carried. Controller processes the incoming cell to determine whether it is the first cell of the packet. Controller might make this determination from information provided in the header (FIG. 3B) of the ATM cell, or from information contained within payload)).

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invitation to implement the method of Yuan within the method of Besset to provide enhanced buffer management in the protocol communication technique in a radio communication network. The new method improves networking performance with discarding redundant data packets.

**Regarding claim 2,** Besset in view of Yuan teaches all the limitations of clam 1. Besset further teaches the method wherein said identifying step comprises the steps of: identifying said next data packet segment as a first data packet segment of said complete data, packet in said buffer if said size of said data packet segment is smaller than said size of said next data packet segment (**[Col 10, lines 36-42)** (e.g. The algorithm (i.e. User defined rules for data packet based on ID comparison) starts by reading the UUI field of a CPS packet being processed. As explained above, this UUI field has the value 27 if at least one other CPS packet follows to complete the AAL2 SDU frame, and the value 26 if, on the contrary, the CPS packet is **the last**

**one of the frame**. Note that a packet can be both the **first** and **last** packet of a frame, as in the case of packet 5 in FIG. 2 (i.e. next data packet segment as a first data packet segment)));

and associating said identified first data packet segment with a first segment identifier ([**Col 3, lines 12-15**] (e.g. In the case of only one CPS PDU, there is also a CPS-PDU start field (STF) octet which serves to **identify the position of the first CPS packet inside the payload** (I.e. identified first data packet) of the ATM cell or CPS PDU**)).**

**Regarding claim 3,** Besset in view of Yuan teaches all the limitations of clam 1. Besset further teaches the method wherein said identifying step comprises the steps of: identifying said next data packet segment as a last data packet segment of said complete data packet in said buffer if said size of said data packet segment differs from said size of said next data packet segment ([**Col 3, lines 65-67, Col 4, lines 1-2**] (e.g. The thus-multiplexed CPS PDUs are then transferred to the ATM layer via the ATM service access point (SAP), The ATM layer comprises ATM cells which are each composed of a 5 octet header (i.e. **header is use for identifying beginning of a data packet** (or next data packet segment) and a 48 octet payload. A CPS PDU can thus be integrally contained in the payload of an ATM cell));

And associating said identified last data packet segment with a last segment identifier ([**Col 2, lines 62-68**] (e.g. a value equal "26" indicates the reception of the **last data** of an SSSAR SDU (i.e. identified last data packet). Any other value, i.e. between 0 and 26 should not be used for reasons of compatibility with other SSCS specifications)).

**Regarding claim 4,** Besset in view of Yuan teaches all the limitations of clam 2. Besset further teaches the method wherein said discarding step comprises the step of discarding said data packet segment associated with said first segment identifier, said data packet segment

associated with said last segment identifier and any intermediate data packet segments between said data packet segment associated with said first segment identifier and said data packet segment associated with said last segment identifier in said buffer (**[Col 11, lines 23-35**) (e.g. If the buffer is saturated, the **algorithm** (i.e. user defined rules to discard the FIRST, LAST or intermediate data from the buffer) follows branch which leads to the **step of discarding the current CPS packet** and setting the parameter PD.sub.AAL2 =TRUE. Consequently, the algorithm for all the following CPS packets up to but excluding the next first packet will then evolve described above to step and thereafter follow branch, whereupon these packets shall be systematically discarded. Thus in the event of a buffer overflow as determined by the decision algorithm for overflow management the current CPS packet and all the following CPS packets of an AAL2 SDU are **systematically discarded even if one or several initial packet(s) had been recorded in the buffer memory**)).

**Regarding claim 5**, Besset teaches a system for managing a data buffer including a queue of consecutive data packets in a base station system of a mobile communications system electronic circuitry configured to (**[Col 7, lines 12-14)** (e.g. FIG. 4 shows the **queuing** of CPS packets at the buffer entrance for the case of example 2, waiting to be multiplexed to form CPS PDUs);

Compare a size of a data packet with a size of a next consecutive data packet in said buffer (**[Col 2, lines 4-12]** (e.g. an ATM adaptation layer comprises a number of sub-layers: a **service specific segmentation** (i.e. a size of a data packet segment with a size of a next consecutive data packet segment in a buffer) and reassembly sub layer (SSSAR) forming part of SEG-SSCS which allows data packets exceeding 45 octets **(i.e. size)** to be segmented));

Identify said complete data packet based on said comparison (**[Col 10, lines 36-42**) (e.g. the algorithm (i.e. based on ID comparison) starts by reading the UUI field of a CPS packet being processed. As explained above, this UUI field has the value 27 if at least one other CPS packet follows to complete the AAL2 SDU frame, and the value 26 if, on the contrary, the CPS packet is the last one of the frame. Note that a packet can be both the first and last packet of a frame, as in the case of packet 5 in FIG. 2 (i.e. next data packet segment as a first data packet segment)));

And discard said identified complete data packet from said buffer (**[Col 11, lines 23-35**) (e.g. If the buffer is saturated, the algorithm follows branch which leads to the **step of discarding the current CPS packet** and setting the parameter PD.sub.AAL2 =TRUE. Consequently, the algorithm for all the following CPS packets up to but excluding the next first packet will then evolve described above to step and thereafter follow branch, whereupon these **packets shall be systematically discarded**. Thus in the event of a buffer overflow as determined by the decision algorithm for overflow management the current CPS packet and all the following CPS packets of an AAL2 SDU are systematically discarded even if one or several initial packet(s) had been recorded in the buffer memory));

Besset differ from the claimed invention in not specifically teaching segments of data packets in a protocol.

However, the preceding limitation is known in the art of communications. In the same field of endeavor, Yuan teaches a method for segmenting data packets (**[Col 1,** lines 10-17] (e.g. Cell relay systems, such as asynchronous transfer mode (ATM) systems, transmit data over a network as a plurality of fixed-length cells. The individual transmissions typically include one or

more cells that constitute a portion of variable-length packets used by end systems or applications. Before transmission, a source station **segments a packet into one or more cells and then transmits the cells** (i.e. A complete data packet is build and transmitted).

Yuan also teaches segments discarding process ([**Col 3,** lines 55-65] (e.g. FIGS. 5A and 5B are flowcharts of **cell discard processing** consistent with the present invention. The processing begins when an output port receives a cell of an incoming packet over a virtual circuit at queuing buffer. Each packet has an **identifier** such as a Virtual Circuit Identifier (VCI), for example, to identify the virtual circuit over which it is carried. Controller processes the incoming cell to determine whether it is the first cell of the packet. Controller might make this determination from information provided in the header (FIG. 3B) of the ATM cell, or from information contained within payload)).

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invitation to implement the method of Yuan within the method of Besset to provide enhanced buffer management in the protocol communication technique in a radio communication network. The new method improves networking performance with discarding redundant data packets.

**Regarding claim 6,** Besset in view of Yuan teaches all the limitations of clam 5. Besset further teaches the system wherein said electronic circuitry is configured to identify said next data packet segment as a first data packet of said complete data packet in said buffer if said size of said data packet segment is smaller than said size of said next data packet segment, said system further comprises means for associating said identified first data packet segment with a first segment identifier (**[Col 10, lines 36-42**) (e.g. The algorithm (i.e. User defined rules for data packet segments based on ID comparison) starts by reading the UUI field of a CPS packet

being processed. As explained above, this UUI field has the value 27 if at least one other CPS packet follows to complete the AAL2 SDU frame, and the value 26 if, on the contrary, the CPS packet is the last one of the frame. Note that a packet can be both the first and last packet of a frame, as in the case of packet 5 in FIG. 2. (I.e. next data packet segment as a first data packet segment));

Said system further comprises means for associating said identified first data packet segment with a first segment identifier **([Col 3, lines 12-15]** (e.g. In the case of only one CPS PDU, there is also a CPS-PDU start field (STF) octet which serves to **identify the position of the first CPS packet inside the payload** (I.e. identified first data packet) of the ATM cell or CPS PDU)**).**

**Regarding claim 7,** Besset in view of Yuan teaches all the limitations of clam 5. Besset further teaches the system wherein said electronic circuitry configured to identify said next data packet segment as a last data packet segment of said complete data packet in said buffer if said size of said data packet segment differs from said size of said next data packet segment, **([Col 10, lines 36-42**) (e.g. The algorithm (i.e. User defined rules for data packet segments based on ID comparison) starts by reading the UUI field of a CPS packet being processed. As explained above, this UUI field has the value 27 if at **least one other** CPS packet follows to complete the AAL2 SDU frame, and the value 26 if, on the contrary, the CPS packet **is the last one of the frame**. Note that a packet can be both the first and last packet of a frame, as in the case of packet 5 in FIG. 2 (i.e. next data packet segment as a first data packet segment)));

said system further comprises means for associating said identified last data packet segment with a last segment identifier (**[Col 2, lines 62-68]** (e.g. A value equal "26" indicates the reception of the **last data** of an SSSAR SDU (i.e. identified last data packet). Any other value, i.e. between 0 and 26 should not be used for reasons of compatibility with other SSCS specifications)).

**Regarding claim 8,** Besset in view of Yuan teaches all the limitations of clam 6. Besset further teaches the system wherein said electronic circuitry is configured to discard said data packet segment associated with said first segment identifier, said data packet segment associated with said last segment identifier and any intermediate data packet segments between said data packet segment associated with said first segment identifier and said data packet segment associated with said last segment identifier in said buffer (**[Col 4, lines 34-43]** (e.g., the present invention proposes a method of managing data packets **originating** from data frames (i.e. **packet segment (FIRST)** , the packets being presented to buffer means prior to processing, wherein, when a packet corresponding to the **start of a frame** (i.e. FIRST) is presented to the buffer stage, it is determined whether the filling level of the buffer means exceeds a first predetermined filling threshold corresponding to a state of congestion and, if such is the case, this packet and all packets belonging to that same **frame are systematically discarded** (i.e. discarding step comprises the step of discarding data based on industry known techniques such as FIFO)).

**Regarding claim 9,** Besset in view of Yuan teaches all the limitations of clam 5. Besset further teaches a base station network node of a base station system in a mobile communications system comprising: a data buffer comprising a queue of consecutive segments of data packets;

([**Col 1, lines 5-11**] (e.g. The present invention relates to the field of data transfer in the form of asynchronous packets in **radio or cable communication networks** (I.e. network with base stations). When packets are transferred asynchronously, there is a risk of congestion at the **network nodes** if the instantaneous flow rate of incoming packets exceeds the maximum throughput capacity of the multiplex));

and a system for managing said data buffer ([**Col 1, lines 12-15**] (e.g. In this context, the invention concerns more specifically--but not exclusively--**a protocol for managing such congestions in buffers** which multiplex network connections known as AAL2 (ATM (asynchronous transfer mode) adaptation layer 2) connections));

**Regarding claim 10**, Besset teaches a method of enabling identification of a complete data packet in a data buffer comprising a queue of consecutive data packet, comprising the steps of: comparing a size of a data packet with a size of a next consecutive data packet in said buffer ([**Col 7, lines 12-14**) (e.g. FIG. 4 shows the **queuing** of CPS packets at the buffer entrance for the case of example, waiting to be multiplexed to form CPS PDUs);

comprising the steps of: said base station system comparing a size of a data packet segment with a size of a next consecutive data packet segment in said buffer ([**Col 2, lines 4-12**] (e.g. an ATM adaptation layer comprises a number of sub-layers: a **service specific segmentation** (i.e. comparing a size of a data packet segment with a size of a next consecutive data packet segment in a buffer ) and reassembly sub layer (SSSAR) forming part of SEG-SSCS which allows data packets exceeding 45 octets **(i.e. size)** to be segmented));

And identifying said complete data packet based on said comparison **([Col 10, lines 36-42)** (e.g. the **algorithm** (i.e. based on ID comparison) starts by **reading** the UUI field of a CPS

packet being processed (i.e. identifying a data packet). As explained above, this UUI field has the value 27 if at least one other CPS packet follows to complete the AAL2 SDU frame, and the value 26 if, on the contrary, the CPS packet is the last one of the frame. Note that a packet can be both the first and last packet of a frame, as in the case of packet 5 in FIG. 2 (i.e. next data packet segment as a first data packet segment))).

Besset differ from the claimed invention in not specifically teaching segments of data packets in a protocol.

However, the preceding limitation is known in the art of communications. In the same field of endeavor, Yuan teaches a method for segmenting data packets ([**Col 1,** lines 10-17] (e.g. Cell relay systems, such as asynchronous transfer mode (ATM) systems, transmit data over a network as a plurality of fixed-length cells. The individual transmissions typically include one or more cells that constitute a portion of variable-length packets used by end systems or applications. Before transmission, a source station **segments a packet into one or more cells and then transmits the cells** (i.e. A complete data packet is build and transmitted).

Yuan also teaches segments discarding process ([**Col 3,** lines 55-65] (e.g. FIGS. 5A and 5B are flowcharts of **cell discard processing** consistent with the present invention. The processing begins when an output port receives a cell of an incoming packet over a virtual circuit at queuing buffer. Each packet has an **identifier** such as a Virtual Circuit Identifier (VCI), for example, to identify the virtual circuit over which it is carried. Controller processes the incoming cell to determine whether it is the first cell of the packet. Controller might make this determination from information provided in the header (FIG. 3B) of the ATM cell, or from information contained within payload)).

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invitation to implement the method of Yuan within the method of Besset to provide enhanced buffer management in the protocol communication technique in a radio communication network. The new method improves networking performance with discarding redundant data packets.

**Regarding claim 11,** Besset in view of Yuan teaches all the limitations of clam 10. Besset further teaches the method further comprising the step of providing a segment counter associated with a data packet segment in said buffer (**[Col 6, lines 28-41]** (e.g. In order to manage the possible congestion and overflow conditions of the buffer efficiently, the method advantageously operates so that: for each incoming CPS packet, the Length Indicator field (LI) of the CPS Packet header is read to determine the length of the arriving CPS Packet; the thus determined length of the arriving CPS packet is used to update a **buffer occupancy counter** (i.e. counter associated with a data packet) which is configured to store at least one amongst: i) the instantaneous number of octets used in the buffer for a given AAL2 connection; ii) the instantaneous number of octets used in the buffer for a given group of AAL2 connections; iii) the instantaneous number of octets used in the buffer by all AAL2 connections stored therein.)).

**Regarding claim 12,** Besset in view of Yuan teaches all the limitations of clam 11. Besset further teaches the method further comprising the steps of:

comparing a size of said data packet segment associated with said counter with a size of a next consecutive data packet segment in said buffer (**[Col 10, lines 36-42)** (e.g. The algorithm (i.e. User defined rules for data packet based on ID comparison) starts by reading the UUI field of a CPS packet being processed. As explained above, this UUI field has the value 27 if at least one other CPS packet follows to complete the AAL2 SDU frame, and the value 26 if, on the contrary,

the CPS packet is **the last one of the frame**. Note that a packet can be both the **first** and **last** packet of a frame, as in the case of packet 5 in FIG. 2 (i.e. next data packet segment as a first data packet segment)));

identifying said next data packet segment as a first data packet segment of said complete data packet in said buffer if said size of said data packet segment associated with said counter is smaller than said size of said next data packet segment (**[Col 2, lines 34-40]** (e.g. the SSSAR will segment (and reassemble) the SSSAR SDUs (in the present case 1 AAL2 SDU=1 SSSAR SDU) into packets referred to as SSSAR protocol data units PDUs whose maximum size by default is 45 octets (i.e. data segment size). Consequently, the 150-octet (e.g. packet **size)** AAL2 SDU of the example can be divided up into three 45-octet SSSAR PDUs (e.g. **First data packet segment**, second data packet segment, Next data packet segment, etc) and the 15 remaining octets (i.e. compare and identify **smaller size packets))** ) can be loaded into a fourth SSSAR PDU)).

**Regarding claim 13,** Besset in view of Yuan teaches all the limitations of clam 12. Besset further teaches the method further comprising the steps of:

(a) comparing a size of the data packet segment currently associated with said counter with a size of a next consecutive data packet segment in said buffer (**[Col 6, lines 28-41]** (e.g. In order to manage the possible congestion and overflow conditions of the buffer efficiently, the method advantageously operates so that: for each incoming CPS packet, the **Length Indicator field** (LI) of the CPS Packet **header** is read to **determine** the **length** of the arriving CPS Packet (e.g. comparing a size of the data packet segment));

and (b) associating said counter with said next data packet segment if said size of the data packet segment currently associated with said counter is equal to or larger than said size of said next data packet segment (**[Col 6, lines 28-41]** (e.g. the thus **determined length** of the arriving CPS packet is used to **update a buffer occupancy counter** (i.e. segment counter associated with a data packet segment) which is configured to store at least one amongst));

and repeating both said comparison step (a) and said associating step (b) until said size of the data packet currently associated with said counter is smaller than said size of said next data packet segment , whereby said next data packet segment is identified as a first data packet segment of said complete data packet in said buffer (**[Col 2, lines 34-40]** (e.g. the SSSAR will segment (and reassemble) the SSSAR SDUs (in the present case 1 AAL2 SDU=1 SSSAR SDU) into packets referred to as SSSAR protocol data units PDUs whose maximum size by default is 45 octets (i.e. **size of a buffer).** Consequently, the 150-octet (e.g. buffer **size )** AAL2 SDU of the example can be divided up into three 45-octet SSSAR PDUs (e.g. First data packet segment, second data packet segment, Next data packet segment, etc) and the 15 remaining octets (i.e. compare and identify **smaller size packets))** ) can be loaded into a fourth SSSAR PDU)).

**Regarding claim 14,** Besset in view of Yuan teaches all the limitations of clam 12. Besset further teaches the method further comprising the step of associating said segment counter with said first data packet segment of said complete data packet (**[Col 6, lines 28-41]** (e.g. In order to manage the possible congestion and overflow conditions of the buffer efficiently, the method advantageously operates so that: for each **incoming** CPS packet (i.e. associating first incoming segment to segment counter with said first data packet segment (FIRST), the Length Indicator field (LI) of the CPS Packet **header** is read to **determine** the

**length** of the arriving CPS Packet; thus determined length of the arriving CPS packet is used to update a **buffer occupancy counter** (i.e. segment counter associated with a data packet segment )).

**Regarding claim 15,** Besset in view of Yuan teaches all the limitations of clam 14. Besset further teaches the method further comprising the steps of:

comparing a size of said data packet segment associated with said counter with a size of a next consecutive data packet segment in said buffer (**[Col 2, lines 34-40]** (e.g. the SSSAR will segment (and reassemble) the SSSAR SDUs (in the present case 1 AAL2 SDU=1 SSSAR SDU) into packets referred to as SSSAR protocol data units PDUs whose maximum size by default is 45 octets (i.e. data segment size). Consequently, the 150-octet (e.g. packet **size)** AAL2 SDU of the example can be divided up into three 45-octet SSSAR PDUs (e.g. **First data packet segment**, second data packet segment, **LAST** data packet segment, etc) and the 15 remaining octets (i.e. compare and identify **smaller size packets))** ) can be loaded into a fourth SSSAR PDU)).

and identifying said next data packet segment as a last data packet segment of said complete data packet in said buffer if said size of said data packet segment associated with said counter differs from said size of said next data packet segment (**[Col 2, lines 62-68]** (e.g. A value equal "26" indicates the reception of the **last data** of an SSSAR SDU (i.e. identified last data packet). Any other value, i.e. between 0 and 26 should not be used for reasons of compatibility with other SSCS specifications)).

**Regarding claim 16,** Besset in view of Yuan teaches all the limitations of clam 15. Besset further teaches the method wherein said complete data packet is identified as comprising

said first data packet segment of said complete data packet, said last data packet segment of said complete data packet and any intermediate data packet segments between said first and last data packet segment of said complete data packet in said buffer (**[Col 2, lines 34-40]** (e.g. the SSSAR will segment (and reassemble) the SSSAR SDUs (in the present case 1 AAL2 SDU=1 SSSAR SDU) into packets referred to as SSSAR protocol data units PDUs whose maximum size by default is 45 octets (i.e. **Segment size ).** Consequently, the 150-octet (e.g. Data Packet **size )** AAL2 SDU of the example can be divided up into three 45-octet SSSAR PDUs (e.g. First data packet segment, intermediate data packet segment, **LAST** data packet segment, etc) )).

**Regarding claim 17,** Besset in view of Yuan teaches all the limitations of clam 15. Besset further teaches the method further comprising the steps of: determining a total size of said first data packet segment of said complete data packet, said last data packet segment of said complete data packet and any intermediate data packet segments between said first and last data packet segment of said complete data packet in said buffer ; comparing said total size with a minimum size threshold; and identifying said complete data packet as comprising said first data packet segment of said complete data packet, said last data packet segment of said complete data packet and any intermediate data packet segments between said first and last data packet segment of said complete data packet in said buffer if said total size is larger than said minimum size threshold (**[Col 8, lines 32-44]** (e.g. This is the **counter which indicates** (i.e. associating with segment) the **buffer's current filling level** (for an AAL2 connection, a group of AAL2 connections or the entire buffer). In other words it indicates the number of octets (i.e. .segments) present in the buffer for a given AAL2 connection (or for a given group of connections, or the

**total number of octets used (i.e.** determining a total size of first (or last) data packet or segment))).

said system further comprises means for associating said identified last data packet segment with a last segment identifier buffer (**[Col 2, lines 62-68]** (e.g. A value equal "26" indicates the reception of the **last data** of an SSSAR SDU (i.e. identified last data packet). Any other value, i.e. between 0 and 26 should not be used for reasons of compatibility with other SSCS specifications)).

**Regarding claim 18,** Besset in view of Yuan teaches all the limitations of clam 11. Besset further teaches the method further comprising the steps of:

comparing a size of said data packet segment associated with said counter with a size of a next consecutive data packet segment in said buffer (**[Col 8, lines 55-65]** (e.g. It is then determined whether the buffer 40 is in a **state of congestion** (step S58). If the buffer is in a state of congestion (CONG_VAL=OK)--as determined at a previous sampling--the algorithm passes along branch b1, and the value CPS_CO+LI+1+3 is **compared with the lower threshold** (step S60). If CPS_CO+LI+1+3 is lower than that threshold 44, then it is **determined that the initially declared congestion condition is no longer true and the algorithm passes along branch** b2 to set the value CONG_VAL=NOK (i.e. comparing a size of said data packet segment ). If CPS_CO+LI+1+3 is greater than or equal to CPS_Low_Threshold, it is determined that a congestion condition still exists and the algorithm passes along branch b3 to maintain CONG_VAL=OK));

and identifying said next data packet segment as a last data packet segment of said complete data packet in said buffer if said size of said data packet segment associated with said

counter differs from said size of said next data packet segment **([Col 10, lines 36-42**) (e.g. The algorithm (i.e. User defined rules for data packet segments based on ID comparison) starts by reading the UUI field of a CPS packet being processed. As explained above, this UUI field has the value 27 if at least one other CPS packet follows to complete the AAL2 SDU frame, and the value 26 if, on the contrary, the CPS packet is the last one of the frame. Note that a packet can be both the first and last packet of a frame, as in the case of packet 5 in FIG. 2 (i.e. next data packet segment as a first data packet segment)));

**Regarding claim 19,** Besset in view of Yuan teaches all the limitations of clam 11. Besset further teaches the method further comprising the steps of:

(c) comparing a size of the data packet segment currently associated with said counter with a size of a next consecutive data packet segment in said (**[Col 8, lines 55-65]** (e.g. It is then determined whether the buffer 40 is in a **state of congestion** (step S58). If the buffer is in a state of congestion (CONG_VAL=OK)--as determined at a previous sampling--the algorithm passes along branch b1, and the value CPS_CO+LI+1+3 is **compared with the lower threshold** (step S60). If CPS_CO+LI+1+3 is lower than that threshold 44, then it is **determined that the initially declared congestion condition is no longer true and the algorithm passes along branch** b2 to set the value CONG_VAL=NOK (i.e. comparing a size of said data packet segment ). If CPS_CO+LI+1+3 is greater than or equal to CPS_Low_Threshold, it is determined that a congestion condition still exists and the algorithm passes along branch b3 to maintain CONG_VAL=OK));

(d) associating said counter with said next data packet segment if said size of the data packet segment currently associated with said counter is equal to said size of said next data

packet segment ; and repeating both said comparison step (c) and said associating step (d) until said size of the data packet segment currently associated with said differs from said size of said next data packet segment, whereby said next data packet segment is identified as a last data packet segment of said complete data packet in said buffer (**[Col 6, lines 28-41]** (e.g. In order to manage the possible congestion and overflow conditions of the buffer efficiently, the method advantageously operates so that: for each incoming CPS packet, the Length Indicator field (LI) of the CPS Packet header is read to determine the length of the arriving CPS Packet; the thus determined length of the arriving CPS packet is used to update a **buffer occupancy counter** (i.e. counter associated with a data packet) which is configured to store at least one amongst: i) the instantaneous number of octets used in the buffer for a given AAL2 connection; ii) the instantaneous number of octets used in the buffer for a given group of AAL2 connections; iii) the instantaneous number of octets used in the buffer by all AAL2 connections stored therein.)).

**Regarding claim 20,** Besset teaches a system for enabling identification of a complete data packet in a data buffer comprising a queue of consecutive data packet (**[Col 7, lines 12-14**) (e.g. FIG. 4 shows the **queuing** of CPS packets at the buffer entrance for the case of example 2, waiting to be multiplexed to form CPS PDUs),

comprising: means for comparing a size of a data packet segment with a size of a next consecutive data packet segment in said buffer and means for identifying said complete data packet based on said comparison (**[Col 8, lines 55-65]** (e.g. It is then determined whether the buffer 40 is in a **state of congestion** (step S58). If the buffer is in a state of congestion (CONG_VAL=OK)--as determined at a previous sampling--the algorithm passes along branch

b1, and the value CPS_CO+LI+1+3 is **compared with the lower threshold** (step S60). If CPS_CO+LI+1+3 is lower than that threshold 44, then it is **determined that the initially declared congestion condition is no longer true and the algorithm passes along branch** b2 to set the value CONG_VAL=NOK (i.e. comparing a size of said data packet segment ). If CPS_CO+LI+1+3 is greater than or equal to CPS_Low_Threshold, it is determined that a congestion condition still exists and the algorithm passes along branch b3 to maintain CONG_VAL=OK));

Besset differ from the claimed invention in not specifically teaching segments of data packets in a protocol.

However, the preceding limitation is known in the art of communications. In the same field of endeavor, Yuan teaches a method for segmenting data packets ([**Col 1,** lines 10-17] (e.g. Cell relay systems, such as asynchronous transfer mode (ATM) systems, transmit data over a network as a plurality of fixed-length cells. The individual transmissions typically include one or more cells that constitute a portion of variable-length packets used by end systems or applications. Before transmission, a source station **segments a packet into one or more cells and then transmits the cells** (i.e. A complete data packet is build and transmitted).

Yuan also teaches segments discarding process ([**Col 3,** lines 55-65] (e.g. FIGS. 5A and 5B are flowcharts of **cell discard processing** consistent with the present invention. The processing begins when an output port receives a cell of an incoming packet over a virtual circuit at queuing buffer. Each packet has an **identifier** such as a Virtual Circuit Identifier (VCI), for example, to identify the virtual circuit over which it is carried. Controller processes the incoming cell to determine whether it is the first cell of the packet. Controller might make this

determination from information provided in the header (FIG. 3B) of the ATM cell, or from information contained within payload)).

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invitation to implement the method of Yuan within the method of Besset to provide enhanced buffer management in the protocol communication technique in a radio communication network. The new method improves networking performance with discarding redundant data packets.

**Regarding claim 21,** Besset in view of Yuan teaches all the limitations of clam 20. Yuan further teaches the system comprising means for associating a segment counter with a data packet segment in said buffer ([**Col 4,** lines 26-39] (e.g. Controller determines the number of cells yet to be received from the total length of the packet. To accomplish this determination, controller contains a mechanism to determine the number of remaining cells that have yet to arrive at buffer. For example, the cell may carry a **decrementing counter field** that explicitly conveys the number of remaining cells. To implement this feature, the system may use the header CRC field 326 (FIG. 3B), for example, as a **counter to convey the length of the packet in terms of cells**. This limits the length of the packet, however, to 256 cells or 12,288 bytes. Alternatively, controller might **contain a counter that decrements** when a cell belonging to the same packet and identified with the same VCI has been stored in buffer (i.e. a system comprising a segment counter with a data packet segment)).

**Regarding claim 22,** Besset in view of Yuan teaches all the limitations of clam 21. Yuan further teaches the system wherein said comparison means is adapted for comparing a size of said data packet segment associated with said counter with a size of a next consecutive data packet segment in said buffer, wherein said identifying means is adapted for identifying said

next data packet segment as a first data packet segment of said complete data packet in said buffer if said size of said data packet segment associated with said counter is smaller than said size of said next data packet segment (**[Col 4,** lines 26-39] (e.g. Controller determines the number of cells yet to be received from the total length of the packet. To accomplish this determination, controller contains a mechanism to determine the number of remaining cells that have yet to arrive at buffer. For example, the cell may carry a **decrementing counter field** that explicitly conveys the number of remaining cells. To implement this feature, the system may use the header CRC field 326 (FIG. 3B), for example, as a **counter to convey the length of the packet in terms of cells**. This limits the length of the packet, however, to 256 cells or 12,288 bytes. Alternatively, controller might **contain a counter that decrements** when a cell belonging to the same packet and identified with the same VCI has been stored in buffer (i.e. a system comparing a size of said data packet segment associated with counter with a size of a next consecutive data packet segment in a buffer)).

**Regarding claim 23,** Besset in view of Yuan teaches all the limitations of clam 21. Besset further teaches the system wherein said comparison means is adapted for comparing a size of the data packet segment currently associated with said counter with a size of a next consecutive data packet segment in said buffer, wherein said associating means is adapted for associating said counter with said next data packet segment if said size of the data packet segment currently associated with said counter is equal to or larger than said size of said next data packet segment, said comparison means is adapted for repeating said size comparison and said associating means is adapted for repeating said counter association until said size of the data packet segment currently associated with said counter is smaller than said size of said next data

packet segment whereby said identifying means is adapted for identifying said next data packet segment as a first data packet segment of said complete data packet in said buffer ([**Col 4,** lines 26-39] (e.g. Controller determines the number of cells yet to be received from the total length of the packet. To accomplish this determination, controller contains a mechanism to determine the number of remaining cells that have yet to arrive at buffer. For example, the cell may carry a **decrementing counter field** that explicitly conveys the number of remaining cells. To implement this feature, the system may use the header CRC field 326 (FIG. 3B), for example, as a **counter to convey the length of the packet in terms of cells**. This limits the length of the packet, however, to 256 cells or 12,288 bytes. Alternatively, controller might **contain a counter that decrements** when a cell belonging to the same packet and identified with the same VCI has been stored in buffer (i.e. a system comparing a size of said data packet segment associated with counter with a size of a next consecutive data packet segment in a buffer)).

**Regarding claim 24,** Besset in view of Yuan teaches all the limitations of clam 22. Besset further teaches the system wherein said associating means is adapted for associating said segment counter with said first data packet segment (FIRST)) of said complete data packet ([**Col 8, lines 32-44**] (e.g. This is the **counter which indicates** (i.e. associating with segment) the **buffer's current filling level** (for an AAL2 connection, a group of AAL2 connections or the entire buffer). In other words it indicates the number of octets (i.e. .segments) present in the buffer for a given AAL2 connection (or for a given group of connections, or the total number of octets used))).

**Regarding claim 25,** Besset in view of Yuan teaches all the limitations of clam 24. Besset further teaches the system wherein said comparison means is adapted for comparing a

size of said data packet segment associated with said counter with a size of a next consecutive data packet segment in said buffer , wherein said identifying means is adapted for identifying said next data packet segment as a last data packet segment of said complete data packet in said buffer if said size of said data packet segment associated with said counter differs from said size of said next data packet segment (**[Col 8, lines 55-65]** (e.g. It is then determined whether the buffer is in a state of congestion. If the buffer is in a state of congestion (CONG_VAL=OK), as determined at a previous sampling, the algorithm passes along branch b1, and the value CPS_CO+LI+1+3 (i.e. size) is **compared** with the lower threshold (i.e. comparing a size of segments or packets) . If CPS_CO+LI+1+3 (i.e. size) is **lower than that threshold**, then it is determined that the initially declared congestion condition is no longer true and the algorithm passes along branch b2 to set the value CONG_VAL=NOK. If CPS_CO+LI+1+3 (i.e. size) is **greater than or equal to** CPS_Low_Threshold, it is determined that a congestion condition still exists and the algorithm passes along branch b3 to maintain CONG_VAL=OK)).

**Regarding claim 26,** Besset in view of Yuan teaches all the limitations of clam 25. Besset further teaches the system wherein said identifying means is adapted for identifying said complete data packet as comprising said first data packet segment of said complete data packet, said last data packet segment of said complete data packet and any intermediate data packet segments between said first and last data packet segment of said complete data packet in said buffer **(Col 4, lines 3-11]** (e.g. In practice, CPS packets are stored in buffers before being transposed into corresponding **ATM cells via the intermediate (**i.e. data packets**)** multiplexing into CPS PDUs. Indeed, the information arrives at the buffers asynchronously and some non-negligible processing time is required to transform the CPS

packets into CPS PDUs. It is thus possible to receive at a given moment more information in the form of CPS packets than can be transferred on the interface. This is the reason why the packets must be buffered)).

**Regarding claim 27,** Besset in view of Yuan teaches all the limitations of clam 25. Besset further teaches the system further comprising means for determining a total size of said first data packet segment of said complete data packet, said last data packet segment) of said complete data packet and any intermediate data packet segments between said first and last data packet segment of said complete data packet in said buffer , said comparison means is adapted for comparing said total size with a minimum size threshold, and said identifying means is adapted for identifying said complete data packet as comprising said first data packet segment (FIRST) of said complete data packet, said last data packet segment of said complete data packet and any intermediate data packet segments between said first and last data packet segment of said complete data packet in said buffer if said total size is larger than said minimum size threshold (**[Col 8, lines 32-44]** (e.g. This is the **counter which indicates** (i.e. associating with segment) the **buffer's current filling level** (for an AAL2 connection, a group of AAL2 connections or the entire buffer). In other words it indicates the number of octets (i.e. .segments) present in the buffer for a given AAL2 connection (or for a given group of connections, or the **total number of octets used (i.e.** determining a total size of first (or last) data packet or segment)).

**Regarding claim 28,** Besset in view of Yuan teaches all the limitations of clam 21. Besset further teaches the system wherein said comparison means is adapted for comparing a size of said data packet segment associated with said counter with a size of a next consecutive

data packet segment in said buffer , wherein said identifying means is adapted for identifying said next data packet segment as a last data packet segment of said complete data packet in said buffer if said size of said data packet segment associated with said counter differs from said size of said next data packet segment (**[Col 8, lines 55-65]** (e.g. It is then determined whether the buffer is in a state of congestion. If the buffer is in a state of congestion (CONG_VAL=OK), as determined at a previous sampling, the algorithm passes along branch b1, and the value CPS_CO+LI+1+3 (i.e. size) is **compared** with the lower threshold (i.e. comparing a size of segments or packets) . If CPS_CO+LI+1+3 (i.e. size) is **lower than that threshold**, then it is determined that the initially declared congestion condition is no longer true and the algorithm passes along branch b2 to set the value CONG_VAL=NOK. If CPS_CO+LI+1+3 (i.e. size) is **greater than or equal to** CPS_Low_Threshold, it is determined that a congestion condition still exists and the algorithm passes along branch b3 to maintain CONG_VAL=OK)).

**Regarding claim 29,** Besset in view of Yuan teaches all the limitations of clam 21. Besset further teaches the system wherein said comparison means is adapted for comparing a size of the data packet segment currently associated with said counter with a size of a next consecutive data packet segment in said buffer, wherein said associating means is adapted for associating said counter with said next data packet segment if said size of the data packet segment currently associated with said counter is equal to said size of said next data packet segment, said comparison means is adapted for repeating said size comparison and said associating means is adapted for repeating said counter associating until said size of the data packet segment currently associated with said counter differs from said size of said next data packet segment , whereby said identifying means is adapted for identifying said next data packet

segment as a last data packet segment (LAST) of said complete data packet in said buffer ([**Col 8, lines 55-65**] (e.g. It is then determined whether the buffer is in a state of congestion. If the buffer is in a state of congestion (CONG_VAL=OK), as determined at a previous sampling, the algorithm passes along branch b1, and the value CPS_CO+LI+1+3 (i.e. size) is **compared** with the lower threshold (i.e. comparing a size of segments or packets) . If CPS_CO+LI+1+3 (i.e. size) is **lower than that threshold**, then it is determined that the initially declared congestion condition is no longer true and the algorithm passes along branch b2 to set the value CONG_VAL=NOK. If CPS_CO+LI+1+3 (i.e. size) is **greater than or equal to** CPS_Low_Threshold, it is determined that a congestion condition still exists and the algorithm passes along branch b3 to maintain CONG_VAL=OK))

**Conclusion**

1. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Mahendra Patel whose telephone number is 571-270-7499. The examiner can normally be reached on 9:30 AM to 5:30 PM EST.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, V. Paul Harper can be reached on 571-272-7605. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http: //pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/MAHENDRA R PATEL/
Examiner, Art Unit 2617

/VINCENT P. HARPER/
Supervisory Patent Examiner, Art Unit 2617